2021 IEEE Green Technologies Conference (GreenTech) | 978-1-7281-9139-3/21/$31.00 ©2021 IEEE | DOI: 10.1109/GREENTECH48523.2021.00079

Trust-based User Interface Design for Islanded Alternating Current Microgrids

Yuliya Matlashova
Electrical and Computer Engineering
University of New Mexico
Albuquerque, NM

Meeko Oishi
Electrical and Computer Engineering
University of New Mexico
Albuquerque, NM

Ali Bidram
Electrical and Computer Engineering
University of New Mexico
Albuquerque, NM

***Abstract*—Microgrid systems can provide extensive information using their measurement units to the operators. As microgrid systems become more pervasive, there will be a need to adjust the information an operator requires to provide an optimized user-interface. In this paper, a combinatorial optimization strategy is used to provide an optimal user-interface for the microgrid operator that selects information for display depending on the operator's trust level in the system, and the assigned task. We employ a method based on sensor placement by capturing elements of the interface as different sensors, that find an optimal set of sensors via combinatorial optimization. However, the typical inverter-based microgrid model poses challenges for the combinatorial optimization due to its poor conditioning. To combat the poor conditioning, we decompose the model into its slow and fast dynamics, and focus solely on the slow dynamics, which are more well conditioned. We presume the operator is tasked with monitoring phase angle and active and reactive power control of inverter-based distributed generators. We synthesize user-interface for each of these tasks under a wide range of trust levels, ranging from full trust to no trust. We found that, as expected, more information must be included in the interface when the operator has low trust. Further, this approach exploits the dynamics of the underlying microgrid to minimize information content (to avoid overwhelming the operator). The effectiveness of proposed approach is verified by modeling an inverter-based microgrid in Matlab.**

***Index Terms*—Distributed generator, human automation, inverter, microgrid, sensor selection, user interface design**

I. Introduction

Microgrids have gained much attention recently because they accommodate reliable and sustainable delivery of power to remote areas and critical power infrastructure. Microgrids are the main building blocks of smart grids, and will play a critical role in increasing the resilience of electric power grids. In a smart power infrastructure, metering and measurement units and sensors are placed all over the system to monitor microgrid in real-time and provide the required information for the governing microgrid control system as well as the supervising human operator [1]–[4].

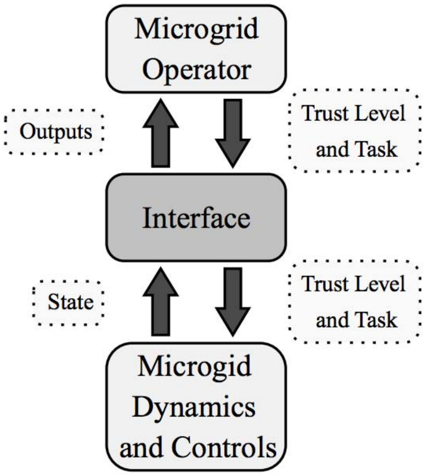

Fig. 1: Operator and microgrid interface for monitoring the system measurements as well as sending the required commands.

The microgrid control is governed by a hierarchical control system that consists of primary, secondary, and tertiary control levels. The microgrid operator oversees the microgrid interaction with the upstream grid, and determines the specific setpoints for the highest control hierarchy, i.e., tertiary control level [5]. With multiple distributed generators (DGs) and loads, the microgrid operator must maintain situational awareness over the entire microgrid, including bus angles, active and reactive power flows from DGs and on the lines, bus voltage magnitudes, and other variables. During extreme events, the operator may experience excessive stress which may impair their decisions in making the right decisions [6], [7].

Information overload is a non-trivial problem, in which the operator is unable to use any of the displayed information, because it sheer volume is over-whelming. Hence, we seek to identify the "right" information for the operator, for different tasks and scenarios. Selection of information for display to the operator must facilitate situational awareness, by enabling

This material is based upon work supported by the National Science Foundation Grant Number OIA-1757207. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. This research was also supported in part by the Laboratory Directed Research and Development program at Sandia National Laboratories, a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA- 0003525. The views expressed in this article do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

2166-5478/21/$31.00 ©2021 IEEE
DOI 10.1109/GreenTech48523.2021.00079

reconstruction and prediction of variables relevant for completion of a desired task. That is, the information displayed must be chosen in accordance with the under-lying dynamics of the microgrid. Further, selection of information is dependent upon the operator's current level of trust in the automation. We presume a feedback linearization structure, in which the automation accomplishes low-level control necessary to complete the desired tasks, so that the operator focuses on high level reference tracking objectives (see Figure 1). An operator that has a high level of trust in the system would be amenable to delegating more authority to the automation, and conversely, an operator with low trust will want to maintain more directed control and supervision.

The main contribution of this paper is the application of a technique for synthesis of optimal dynamics-driven user interfaces that are responsive to the operator's trust in the underlying microgrid automation. The theory for this approach was developed in [8]. However, the typical microgrid model creates numerical difficulties because microgrid dynamics are poorly conditioned, making rank calculations necessary for feedback linear controllers difficult to compute. Hence, we implement a decomposition into fast and slow dynamics to improve numerical conditioning, and a user-interface is designed that optimizes the information an operator needs. This optimization is assured to produce a minimal amount of information for the operator to complete the assigned tasks, while satisfying constraints for situational awareness and trust.

The rest of this paper is organized as follows: Section II covers the microgrid control systems and introduces the microgrid model. In Section III, the operator's tasks are presented and discussed. The interface design framework is presented in Section IV, along with our state decomposition. In Section V, we present our optimization results and discuss implications for microgrids.

II. Preliminaries of Microgrid Control System and Dynamical Model

A. Microgrid Control System

The microgrid control system is implemented through a hierarchical control structure consisting of primary, secondary, and tertiary control levels. The role of the primary control is to facilitate a smooth transition from grid-connected to islanded mode. Primary control is usually implemented through the voltage and frequency droop techniques which have a key role for maintaining voltage and frequency stability of the microgrid after islanding occurs. The secondary control level is responsible for voltage regulation and frequency restoration after the primary control is applied. The secondary control has a slower response compared to primary control and is expected to regulate microgrid's voltage and restore its frequency in less than a minute. Finally, tertiary control is utilized to control the active and reactive power flow between microgrid and upstream grid in the grid-connected mode [5]. Each microgrid can be also supervised and controlled by the microgrid operator that oversees the microgrid interaction with the upstream grid and determines the specific setpoints for the highest control hierarchy, i.e., tertiary control level.

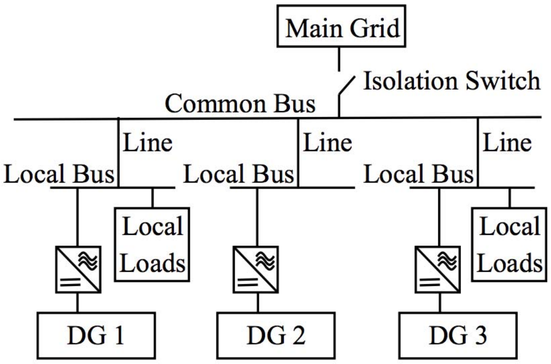

Fig. 2: Structure of an inverter-based microgrid [9]

.

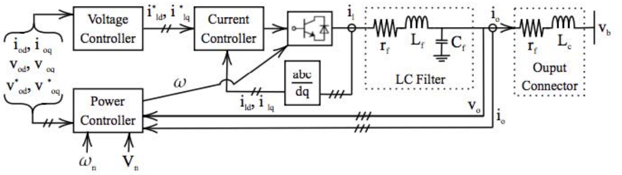

Fig. 3: Individual DG diagram, a closer view of each DG inverter and its control loops.

B. Microgrid Dynamical Model

The microgrid model includes the dynamics of distribution generators, lines, and loads. A typical microgrid system was modeled with three distribution generators (DGs), two lines, and two loads. We consider a microgrid structure from [9] redrawn in Figure 2. The individual DG inverter block diagram are shown in Figure 3, based on the model we use from [9].

1) Inverter-based DG Model: The individual inverter-based DG model is comprised of the internal power, voltage, and current controllers as well as an output inductor-capacitor (LC) filter and output coupling inductance as seen in Figure 3, details are provided in [9]. The DG inverter model can be described by the following equation

$$\begin{aligned}\dot{x}_{invi} \quad &= \quad A_{INVi}\Delta x_{invi} + B_{INVi}\Delta v_{bDQi} \\ &\qquad +B_{iwcom}\Delta w_{com}\end{aligned} \tag{1}$$

where

$$\begin{aligned}\Delta x_{invi} = [\Delta\delta_i \quad &\Delta P_i \quad \Delta Q_i \quad \Delta\phi_{di} \quad \Delta\phi_{qi} \\ &\Delta\gamma_{di} \quad \Delta\gamma_{qi} \quad \Delta i_{ldi} \quad \Delta i_{lqi} \\ &\Delta v_{odi} \quad \Delta v_{oqi} \quad \Delta i_{odi} \quad \Delta i_{oqi}]^\top.\end{aligned} \tag{2}$$

The matrices A_{INVi}, B_{INVi}, and B_{iwcom} from equation (1) are given in [9]. In equation (1) the first three state variables are phase angle $\Delta\delta_i$, active power ΔP_i, and reactive power ΔQ_i. The state variables for the voltage controller are $\Delta\phi_{di}$ and $\Delta\phi_{qi}$. The current controller state variables are $\Delta\gamma_{di}$ and $\Delta\gamma_{qi}$. Then the output inductor currents are Δi_{ldi} and Δi_{lqi}.

The last states are the output voltage and output current of the DG inverter, Δv_{odi}, Δv_{oqi}, and Δi_{odi}, Δi_{oqi}, respectively.

2) Network Model: The network model, consisting of two lines, is described by the following equations for direct and quadrature components of current of the $line$ connecting nodes j and k.

$$\begin{aligned}\dot{i}_{lineD} &= \tfrac{-r_{line}}{L_{line}} i_{lineD} + w i_{lineQ} + \tfrac{1}{L_{line}} v_{bDj} \\ &\quad - \tfrac{1}{L_{line}} v_{bDk} \\ \dot{i}_{lineQ} &= \tfrac{-r_{line}}{L_{line}} i_{lineQ} - w i_{lineD} + \tfrac{1}{L_{line}} v_{bQj} \\ &\quad - \tfrac{1}{L_{line}} v_{bQk}\end{aligned} \tag{3}$$

where i_{lineD} and i_{lineQ} represent the bus current, and v_{bD} and v_{bQ} represent the bus voltage. The line resistance is represented by r_{line}, L_{line} is the line inductance, and ω is the line frequency.

3) Load Model: An resistive/inductive load model is chosen to represent each of the two loads on the microgrid, and is described by

$$\begin{aligned}\dot{i}_{loadD} &= \tfrac{-r_{load}}{L_{load}} i_{loadD} + w i_{loadQ} + \tfrac{1}{L_{load}} v_{bD} \\ \dot{i}_{loadQ} &= \tfrac{-r_{load}}{L_{load}} i_{loadQ} - w i_{loadD} + \tfrac{1}{L_{load}} v_{bQ}\end{aligned} \tag{4}$$

where i_{loadD} and i_{loadQ} represent the load current, and v_{bD} and v_{bQ} represent the load voltage. Then r_{load} represents the load resistance, L_{load} represents the load inductance, and ω is the frequency.

4) Comprehensive Microgrid Model: The comprehensive inverter-based microgrid dynamical model is composed of the inverter, network, and load dynamics and is described by

$$\Delta\dot{x}_{MG} = A_{MG}\Delta x_{MG} + B_{MG}u_{MG} \tag{5}$$

where $\Delta x_{MG} = [\Delta x_{INV}^T, \Delta i_{lineDQ}^T, \Delta x_{loadDQ}^T]^T$. States x_{INV}, i_{lineDQ}, and i_{loadDQ} represent the DG, line, and load dynamics in the D-Q reference frame, respectively. The matrices A_{MG} and B_{MG} (5) are extracted from [9]. The term u_{MG} includes the frequency and voltage droop references of all DGs as described in [9].

III. Microgrid Operator Tasks Description

We consider three tasks, namely, (1) DG inverter phase angle monitoring, (2) active power monitoring, and (3) reactive power monitoring. Within each task we consider sub-tasks that depend on which DG is chosen. These tasks have been selected because they are are all critical in maintaining a healthy microgrid system.

Within each task, we consider that the microgrid operator may have different levels of trust in the system. Trust levels can range from full trust to no trust, along a graded scale which can be correlated with known metrics to assess trust. The operator's trust levels may be dependent on their previous experience in working with a microgrid, their disposition, as well as ongoing, scenario dependent factors. For example, a novice operator may be less willing to fully trust the system, whereas an expert may be comfortable using less information, and relying on the automation as needed.

A. DG Phase Angle Monitoring

We presume the microgrid operator monitors the phase angles (δ_i) of DG2 and DG3 with respect to DG1 as the reference phase angle, separately, to ensure that they are within stables ranges. If one of them violates the stable ranges, this can be considered as sign of microgrid instability and remedial actions would be required to push the microgrid back to a stable operating condition. In the microgrid dynamical model discussed in the previous subsection, we assume that DG1 is the reference DG, and the phase angle of all other DGs are compared with respect to the phase angle of DG1.

B. DG Active Power Monitoring

The next class of tasks is related to active power (P_i) monitoring of each DG. The microgrid operator monitors the active power of each DG, to evaluate the total amount of generation available in the microgrid. The available active power (or real power) generation in a microgrid can play a critical role in providing microgrid frequency stability and ensures the reliable supply of power to the microgrid's customers.

C. DG Reactive Power Monitoring

The final class of tasks related to the reactive power (Q_i) of a grid system. The microgrid operator monitors the reactive power of each of the DGs to evaluate the total amount of reactive power generation available in the microgrid. The available reactive power generation in a microgrid can play a critical role in the microgrid voltage stability.

IV. User interface design via sensor selection

A. MIMO Input-Output Linearization for LTI Systems

An approach was developed in [8] to pose the user-interface design problem as a combinatorial optimization problem, based on methods for sensor selection. For the system (14), consider an output

$$y = C_{\mathcal{S}}\overline{x} \tag{6}$$

with output matrix $C_{\mathcal{S}}$, a matrix whose rows consist of the elements $s_i \in \mathcal{S}$, such that the total number of outputs associated with $C_{\mathcal{S}}$ is $p = |\mathcal{S}|$. We denote the set of all sensors as $\mathscr{S} = \{s_1, \cdots, s_{|\mathscr{S}|}\}$ for a finite $|\mathscr{S}| \in \mathbb{N}$.

We presume that MIMO feedback linearization is used for control, with the same outputs as are available in the interface. In brief, given an output matrix $C_{\mathcal{S}}$, we construct a similarity transform $P_{\mathcal{S}} \in \mathbb{R}^{n\times n}$,

$$\left[\frac{\xi(t)}{\eta(t)}\right] = P_{\mathcal{S}}\overline{x}(t) = \left[\frac{T_{\mathcal{S}}}{T_{\mathcal{S}}^{\perp}}\right]\overline{x}(t) \tag{7}$$

that results in observable states $\xi(t) \in \mathcal{R}(T_{\mathcal{S}})$ and unobservable states $\eta(t) \in \mathcal{R}\left(T_{\mathcal{S}}^{\perp}\right)$. The observable states are synonymous with those sensors presented in the interface. The linear transformation $T_{\mathcal{S}}$ is defined using T_{s_i} for some $s_i \in \mathcal{S}$, as

$$T_{s_i} = \left[s_i \; (s_i^\top A_0)^\top \; (s_i^\top A_0^2)^\top \; \ldots \; (s_i^\top A_0^{\gamma(s_i)-1})^\top \right]^\top, \quad (8)$$

$$T_{\mathcal{S}} = \text{basis}\left(\mathcal{R}\Big(\Big[T_{s_1}^\top \; T_{s_2}^\top \; \ldots \; T_{s_{|\mathcal{S}|}}^\top \Big] \Big)^\top \right), \quad (9)$$

where $\gamma : \mathscr{S} \rightarrow \mathbb{N}_{[1,n]}$ is the relative degree of the MISO system with the single output $s_i^\top x(t)$, and $\mathcal{R}(\cdot)$ is the range space operator. By (8), $\mathcal{R}(T_{\mathcal{S}})$ is the state subspace spanned by the outputs characterized by $y(t) = C_{\mathcal{S}}\bar{x}(t)$ and its higher derivatives.

B. Fast-slow dynamics

One of the major challenges in implementing the approach above is the numerical ill-conditioning of the microgrid model (28). Every possible set of sensors leads to a different feedback linearized controller, and requires calculations of the rank of (8). Rank operations are known to be non-numerically robust to ill conditioned system matrices. Before proceeding with the approach described, the model is first simplified and transformed, to mitigate the numerical challenges associated with the required rank calculation.

As described in the previous section, the microgrid dynamical model is formulated using d-q reference frame theory. In this model, the reference frame of DG1 is being considered to be the common reference frame. As such, the phase angle of DG1 is always constant and equal to zero. The DG1 phase angle is the first state variable in (5). In order to avoid an unwanted pole at the origin, without loss of generality, (5) can be rewritten as

$$\Delta\dot{x}'_{MG} = A'_{MG}\Delta x'_{MG} + B'_{MG}u_{MG} \quad (10)$$

where x'_{MG} is extracted by removing the first element of x_{MG}; A'_{MG} is derived by removing the first row and column of A_{MG}; B'_{MG} is derived by removing the first row of B_{MG}. The condition number of the model described by (5) was $1.4409e^{22}$, but the simplified model (10) has condition number $1.4357e^{13}$.

We then employ an approach described by Kokotovic and Haddad [10] to separate equation (10) into two time scales, corresponding slow and fast modes of the system.

$$\begin{aligned} \dot{x} &= A_{11}x + A_{12}z + B_1u & (11) \\ \mu\dot{z} &= A_{21}x + A_{22}z + B_2u & (12) \end{aligned}$$

As in [10], when the positive singular perturbation parameter μ is presumed to be very small, (12) becomes

$$0 = A_{21}\bar{x} + A_{22}\bar{z} + B_2\bar{u} \quad (13)$$

Hence, for A_{22} that is invertible, a substitution of $(\bar{z})$ into (11) results in the simplified dynamics

$$\dot{\bar{x}} = A_0\bar{x} + B_0\bar{u} \quad (14)$$

where

$$\begin{aligned} A_0 &= A_{11} - A_{12}{A_{22}}^{-1}A_{21} & (15) \\ B_0 &= B_1 - A_{12}{A_{22}}^{-1}B_2 & (16) \end{aligned}$$

Task Matrices $C_{\mathcal{S}_{\text{task}}}$			
	Phase Angle	Active Power	Reactive Power
DG1	–	e_1^T	e_2^T
DG2	e_{13}^T	e_{14}^T	e_{15}^T
DG3	e_{26}^T	e_{27}^T	e_{28}^T

TABLE I: We consider task matrices, for each task, where $e_i \in \mathbb{R}^{38}$ represents the coordinate vector, with a value 1 in its i^{th} element and zero values elsewhere. All tasks are functions of the approximate slow dynamics (17).

We apply this approach to equation (10), such that

$$A'_{MG} = \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array} \right] \text{ and } B'_{MG} = \left[\begin{array}{c} B_1 \\ \hline B_2 \end{array} \right].$$

The slow mode of (10) is described by the first 38 states, related to the DG inverters of the system, and the fast mode describes the dynamics of last 8 states, related to the lines and loads. We then substitute (14) following (15-16) to obtain

$$\Delta\dot{\bar{x}}'_{MG} = A'_{MG_0}\Delta\bar{x}'_{MG} + B'_{MG_0}\bar{u}_{MG}, \quad (17)$$

a reduced model with 38 states as compared to 47 in the initial model (5). The condition number of A_{MG_0} is $1.5135e^{11}$.

C. Task selection

A task is presumed to be characterized by linear combinations of the state, meaning that the task can be succinctly captured via a task matrix, $C_{\mathcal{S}_{\text{task}}} \in \mathbb{R}^{|\mathcal{S}_{\text{task}}| \times n}$ associated with $\mathcal{S}_{\text{task}} \in 2^{\mathscr{S}}$.

We note that the selected tasks are functions solely of the state of the slow dynamics, so the reduced model (17) is sufficient to capture the tasks of interest. We construct task matrices for each of the tasks and subtasks described in Section 3.

D. Combinatorial optimization for sensor selection

For a given task $C_{\mathcal{S}_{\text{task}}}$, we seek to design a user-interface $C_{\mathcal{S}}$ that satisfies, in order of importance:

1) Necessary conditions for situation awareness,
2) Compatibility with the user's trust in the automation, and
3) Conciseness.

These properties represent human factors that are key for effective human-automation interaction. The first constraint takes into account the limitations of the human operator and the complexity of the task; the second requires that more information is provided to the user when the user's trust in the automation is low, and vice versa; the third constraint prevents high cognitive load associated with excessive data.

The goal is to find a user-interface that satisfies these criteria, given the underlying dynamics of the system. The problem is stated as the following combinatorial optimization problem:

$$\begin{aligned} \underset{\mathcal{S}\in 2^{\mathscr{S}}}{\text{minimize}} \quad & |\mathcal{S}| && \text{(conciseness)} && \text{(18a)} \\ \text{subject to} \quad & \mathcal{S} \in \mathscr{S}_{\text{sit-aware}} && \text{(situation awareness)} && \text{(18b)} \\ & \mathcal{S} \in \mathscr{S}_{\text{trust}} && \text{(trust)} && \text{(18c)} \end{aligned}$$

and can be reduced as in [8] to

$$\begin{aligned} \underset{\mathcal{S}\in 2^{\mathscr{S}}}{\text{minimize}} \quad & |\mathcal{S}| \\ \text{subject to} \quad & \Gamma(\mathcal{S}) = \Gamma(\mathcal{S} \cup \mathcal{S}_{\text{task}}) \\ & \Gamma(\mathcal{S}) \geq k_{\text{trust}} \end{aligned} \tag{19}$$

with user-information index $\Gamma(\mathcal{S}) = \mathcal{R}(T_{\mathcal{S}}) = \text{rank}(T_{\mathcal{S}})$. The user information index $\Gamma(\mathcal{S})$ characterizes the dimensions of $\xi(t)$ and $\eta(t)$, since $\xi(t) \in \mathbb{R}^{\Gamma(\mathcal{S})}$ and $\eta(t) \in \mathbb{R}^{n-\Gamma(\mathcal{S})}$. The observable dynamics $\xi(t)$ are translated to the user-interface, and the unobservable dynamics $\eta(t)$ are hidden. We seek to delegate as much control to the automation as the operator can tolerate, by reducing $\xi(t)$ and therefore increasing $\eta(t)$, as allowed by the dynamics and the human factors constraints.

The trust level k_{trust} could correspond to different trust metrics [11]. Questionnaire-based trust metrics [12]–[14] employ a summative assessment of trust. For example, the SHAPE Automation Trust Index [12] evaluates trust as a percentage (ranging from 0% (no trust) to 100% (full trust). Such a metric could be transformed to our trust level scale, ranging from $\Gamma(\mathscr{S})$ (low trust) to 1 (high trust), respectively, to determine the trust level k_{trust} for a given user and circumstance.

Solutions to (19) are computationally tractable because the user-information index, $\Gamma(\mathcal{S})$, is a submodular and monotone increasing function [8], [15]. For calculations here, we employ an implementation that was used to evaluate an interface for the IEEE 118-bus [8]. This implementation invokes constraint programming in combination with a careful enumeration framework that uses binary variables to succinctly represent various sensor combinations. This helps prevent full enumeration of the entire sensor space.

For low levels of trust, constraint programming is used to solve (19) over a reduced solution space and assures an optimal solution. For high levels of trust, a greedy algorithm is employed for submodular optimization to solve (19), resulting in a suboptimal solution. Similarly, for trust levels in between, a submodular optimization problem is solved to suboptimality with known bounds.

V. Optimization Results

This optimization problem strives to provide the minimal amount of information to an operator while ensuring that that the operator has enough information to complete the task. The information the operator requires to successfully complete their tasks at each trust level should not be overwhelming or force the operator to rely on complex calculations. Doing so may result in the operator losing awareness of the automation and can lead to significant problems.

We implemented the optimization algorithm in MATLAB, running on a standard desktop computer. The computation times for the optimizations were all fast, the slowest computation only took roughly 250ms (Table VII). The computation time was recorded for every task at every trust level (enumerated from 1 to 20). The computation time was averaged over all trust levels, separately for each method implemented, and on average, the computation took no longer than 200ms. This is important as it implies potential for run-time implementation.

Computation Time (s)			
Method	Avg.	Max	Min
Constraint Programming	0.1874	0.1874	0.1874
Submodular Optimization	0.1901	0.2294	0.1548

TABLE II: Computation timetable for phase angle monitoring of DG2. Constraint programming was implemented for $k_{trust} = 1$, and submodular optimization for $k_{trust} = 2$ through $k_{trust} = 38$.

A. Phase Angle

For phase angle monitoring trust levels were enumerated from 1 (high trust) to 20 (low trust). Computation time is shown in Table II and Table III.

1) Phase Angle of DG2: The optimization chose essential sensors at the first DG inverter as well as choosing the sensor related to the task. As k_{trust} increased by 1, meaning there was slight loss of trust (decrease of roughly 5.25%) by the operator, the optimization chose to include a sensor at DG1 (P), suggesting that the active power of DG1 is an essential sensor for the interface. As the operator's trust in the automation continued decreasing (k_{trust} increasing) additional sensors were chosen at the second DG and then the third DG. The optimization chooses to initially select the sensors at points where the relative degree of the corresponding state is 2. When the user is fully distrustful (0% trust) in the automation, an additional sensor is selected at DG1 (v_{oq}, q-axis DG inverter-based voltage). The point of common coupling, where the microgrid connects to the main grid, extends from the same node that DG1 connects to. We speculate that DG1 is fundamentally important for system stability and operator success in completing tasks. Table II shows the computation times for phase angle monitoring of DG2, maximum, minimum and averaged over all trust levels, separately for each method implemented.

2) Phase Angle of DG3: For this task we observed similar patterns. For high trust, the optimization chose sensors at the first DG as well as the sensor related to the task. With lower levels of trust, additional essential sensors were selected at the second and then third DGs. The resulting user-interface for this task was designed almost identically to the one from the previous task, with the exception of the sensor related to the task. Table III shows the computation times for phase angle monitoring of DG3, as well as maximum, minimum and averaged times over all trust levels, separately for each method implemented.

Computation Time (s)			
Method	Avg.	Max	Min
Constraint Programming	0.1140	0.1140	0.1140
Submodular Optimization	0.1767	0.2080	0.1380

TABLE III: Computation timetable for phase angle monitoring of DG3. Constraint programming was implemented for $k_{trust} = 1$, and submodular optimization for $k_{trust} = 2$ through $k_{trust} = 38$.

Computation Time (s)			
Method	Avg.	Max	Min
Constraint Programming	0.09545	0.0992	0.0917
Submodular Optimization	0.1878	0.2304	0.1566

TABLE IV: Computation timetable for active power monitoring of DG1. Constraint programming was implemented for $k_{trust} = 1, 2$, and submodular optimization for $k_{trust} = 3$ through $k_{trust} = 38$.

B. Active Power

Another series of operator tasks related to essential aspects of a microgrid is monitoring the active power of each DG. Trust levels were again enumerated from 1 (high trust) to 20 (low trust). Computation times were fast for these three tasks, shown in Table IV, Table V, and Table VI), respectively, as well.

1) Active Power of DG1: The optimal solutions for active power monitoring included many of the similar sensors chosen for phase angle monitoring. One difference to note is that as k_{trust} increased by 1, the optimization only selected the sensor related to the task ($100\% - 94.5\%$ trust), unlike for phase angle monitoring, in which a second sensor was chosen as soon as trust decreased slightly. As trust decreases, the optimization continues to select additional critical sensors at the first, second, and then third DG inverter, in addition to the sensor related to the task. At very low levels of trust (5.25% trust), a sensor is selected at DG1 (v_{oq}). Then at full loss of trust (0% trust), a final sensor is selected at DG2 for the phase angle. The v_{oq} sensor provides the voltage-source DG inverter voltage on the q-axis. The phase angle of DG2 proves to be an important metric when the tasks are not related to phase angle. Table IV shows the computation time for active power monitoring of DG1, maximum, minimum and averaged over all trust levels, separately for each method implemented.

2) Active Power of DG2: The interface design is very similar to the previous task. Similar patterns of sensor selection by the optimization for the user-interface are observed. The optimization selects the sensor related to the task for $k_{trust} = 1$ and $k_{trust} = 2$, and as the operator continues to lose trust in the system additional sensors are chosen at DG1, then DG2, and then DG3. The same two sensors are selected when the user has very low to no trust in the automation as in the previous task. Table V shows the computation time for active power monitoring of DG2, maximum, minimum and averaged over all trust levels, separately for each method implemented.

Computation Time (s)			
Method	Avg.	Max	Min
Constraint Programming	0.09595	0.1000	0.09190
Submodular Optimization	0.1812	0.2082	0.1495

TABLE V: Computation timetable for active power monitoring of DG2. Constraint programming was implemented for $k_{trust} = 1, 2$, and submodular optimization for $k_{trust} = 3$ through $k_{trust} = 38$.

Computation Time (s)			
Method	Avg.	Max	Min
Constraint Programming	0.09370	0.09440	0.09300
Submodular Optimization	0.1774	0.2038	0.1358

TABLE VI: Computation timetable for active power monitoring of DG3. Constraint programming was implemented for $k_{trust} = 1, 2$, and submodular optimization for $k_{trust} = 3$ through $k_{trust} = 38$.

3) Active Power of DG3: The interface design is very similar to the two previous tasks regarding active power monitoring. The same pattern of sensor selection is followed as previously discussed, with a substitution of the sensor related to the task with the current task. Table VI shows the computation time for active power monitoring of DG3, maximum, minimum and averaged over all trust levels, separately for each method implemented.

C. Reactive Power

The third set of tasks corresponds to monitoring the reactive power of DG1, DG2, and DG3. The trust levels were enumerated from 1 to 20, high trust to low trust, respectively. For the reactive power monitoring of DG3 task in this section, a diagram is provided to portray the user-interface produced by the optimization at three different trust levels. The computation time was fast for these tasks, shown in Table VII, Table VIII, and Table IX, respectively, as well.

1) Reactive Power of DG1: The optimal user-interface for reactive power monitoring contained many of the similar sensors selected for active power monitoring. Other than the sensor that is related to the task being selected, the user-interface is built following the same pattern as for active power monitoring. Table VII shows the computation time for reactive power monitoring of DG1, maximum, minimum and averaged over all trust levels, separately for each method implemented.

2) Reactive Power of DG2: The sensor related to the task is selected for the first two values of k_{trust} before the optimization selects additional sensors from DG1, DG2, and then DG3. The same pattern is followed in sensor selection for this task as it was for active power monitoring. Table VIII shows the computation time for reactive power monitoring of DG2, maximum, minimum and averaged over all trust levels, separately for each method implemented.

3) Reactive Power of DG3: Sensors are chosen in a similar pattern as the were discussed in the active power monitoring

Computation Time (s)			
Method	Avg.	Max	Min
Constraint Programming	0.1655	0.1758	0.1552
Submodular Optimization	0.1933	0.2499	0.1395

TABLE VII: Computation timetable for reactive power monitoring of DG1. Constraint programming was implemented for $k_{trust} = 1, 2$, and submodular optimization for $k_{trust} = 3$ through $k_{trust} = 38$.

Computation Time (s)			
Method	Avg.	Max	Min
Constraint Programming	0.1281	0.1640	0.09220
Submodular Optimization	00.1786	0.2038	0.1404

TABLE VIII: Computation timetable for reactive power monitoring of DG2. Constraint programming was implemented for $k_{trust} = 1, 2$, and submodular optimization for $k_{trust} = 3$ through $k_{trust} = 38$.

Computation Time (s)			
Method	Avg.	Max	Min
Constraint Programming	0.09615	0.09680	0.09550
Submodular Optimization	0.1770	0.2082	0.1403

TABLE IX: Computation timetable for reactive power monitoring of DG3. Constraint programming was implemented for $k_{trust} = 1, 2$, and submodular optimization for $k_{trust} = 3$ through $k_{trust} = 38$.

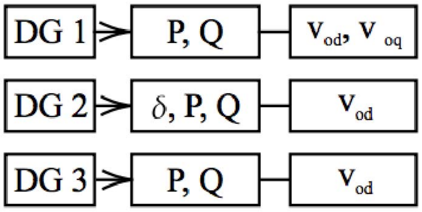

Fig. 4: Interface for the reactive power (Q) monitoring task for DG3, where operator distrusts automation ($k_{trust} = 20$).

cases, other than the sensor selected that related to the task. Table IX shows the computation time for reactive power monitoring of DG3, maximum, minimum and averaged over all trust levels, separately for each method implemented.

A diagram representation of sensor placement chosen by the optimization using three different user trust levels for reactive power monitoring of DG3 is shown in Figs. 4, 5, and 6. In Figure 4 the trust level was chosen to be 20, corresponding to no trust, to capture operator lack of confidence in the automation during an event such as a sudden loss of voltage and frequency stability in the microgrid. For this type of event, we assume the operator will need to take manual control of the automation to restore it to stable working conditions, therefore needing all vital information about the microgrid. An example of such a disturbance could be an unanticipated loss of power generation from one or more DGs due to a severe storm or an accident at the generation source.

A intermediary trust level of 11 was chosen to represent medium trust (Figure 5) which may signify a disturbance the operator is familiar with or one that is not necessarily detrimental, such as an unanticipated spike in power demand. For example, this could be something like a populated event that produces a heavy load on the system, or an industrial business powering up large machines without prior warning to the microgrid operators. We assume these events would cause some caution for the microgrid operator, but with the proper expertise it may be remedied with an intermediary amount of information.

For a high trust scenario (Figure 6), the trust level was chosen to be 3 to represent a scenario in which the operator has a high level of confidence in the system such as when the microgrid operating within the stable ranges with no inherent problems. It may also signify that an expert operator may be overseeing the microgrid's operations. We assume the operator feels confident in the automation controls keeping the microgrid within its stable ranges, therefore the operator requires very little information.

VI. Conclusion

In this paper, a state-space model of a microgrid was introduced composed of three DG inverters and two loads, based on a known model in [9]. Each subsystem was individually modeled and transformed to a common reference frame provided by the first DG inverter to obtain a comprehensive microgrid model. This model was shown to have very poor conditioning and posed complications for the user-interface design optimization. A strategy was chosen where initially a pole at the origin was removed and then the system was decomposed into two time scales, the slow and fast modes. After algebraic manipulation due to removal of a pole at the origin, the system is reduced from a 47 order system to a 38 order system by considering only the slow dynamics. Exploiting the fact that the tasks are related to the slow dynamics of the system, it was possible to continue with this reduced system for the interface optimization.

Future work from this analysis can be expanded to include much larger power systems and with varying combinations of user tasks. For example, the number of distribution generator inverters may be increased in each microgrid. And an example of combining operator tasks would be for the task to be monitoring active power of all distribution generators in the micrgorid, or something of that nature. It is important to note that the optimization took, on average, less than 185 milliseconds to compute using submodular optimization and less than 125 milliseconds for the constraint programming approach. This implies that larger system may have little issue with regards to time constraints in completing the optimization.

References

[1] H. Wang, Z. Yan, M. Shahidehpour, X. Xu, and Q. Zhou, "Quantitative evaluations of uncertainties in multivariate operations of microgrids," *IEEE Transactions on Smart Grid*, 2020.

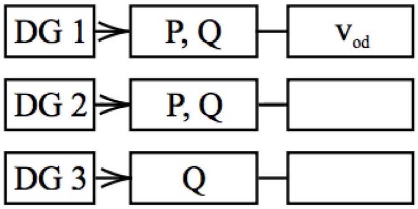

Fig. 5: Interface for the reactive power (Q) monitoring task for DG3, where operator has some distrust in automation ($k_{trust} = 11$).

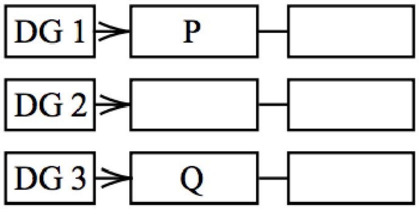

Fig. 6: Interface for the reactive power (Q) monitoring task for DG3, where operator has very high trust in automation ($k_{trust} = 3$).

[2] Q. Zhou, M. Shahidehpour, A. Paaso, S. Bahramirad, A. Alabdulwahab, and A. Abusorrah, "Distributed control and communication strategies in networked microgrids," *IEEE Communications Surveys & Tutorials*, 2020.

[3] A. Mustafa, B. Poudel, A. Bidram, and H. Modares, "Detection and mitigation of data manipulation attacks in ac microgrids," *IEEE Transactions on Smart Grid*, vol. 11, no. 3, pp. 2588–2603, 2019.

[4] B. P. Poudel, A. Mustafa, A. Bidram, and H. Modares, "Detection and mitigation of cyber-threats in the dc microgrid distributed control system," *International Journal of Electrical Power & Energy Systems*, vol. 120, p. 105968, 2020.

[5] A. Bidram and A. Davoudi, "Hierarchical structure of microgrids control system," *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 1963–1976, 2012.

[6] M. Rahnamay-Naeini, Z. Wang, N. Ghani, A. Mammoli, and M. M. Hayat, "Stochastic analysis of cascading-failure dynamics in power grids," *IEEE Transactions on Power Systems*, vol. 29, no. 4, pp. 1767–1779, 2014.

[7] Z. Wang, M. Rahnamay-Naeini, J. M. Abreu, R. A. Shuvro, P. Das, A. A. Mammoli, N. Ghani, and M. M. Hayat, "Impacts of operators' behavior on reliability of power grids during cascading failures," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 6013–6024, 2018.

[8] A. P. Vinod, A. J. Thorpe, P. A. Olaniyi, T. H. Summers, and M. Oishi, "Sensor selection for dynamics-driven user-interface design," *IEEE Transactions on Control Systems Technology*, 2021.

[9] N. Pogaku, M. Prodanovic, and T. C. Green, "Modeling, analysis and testing of autonomous operation of an inverter-based microgrid," *IEEE Transactions on Power Electronics*, vol. 22, no. 2, pp. 613–625, 2007.

[10] P. V. Kokotovic and A. H. Haddad, "Controllability and time-optimal control of systems with slow and fast modes," in *1974 IEEE Conference on Decision and Control including the 13th Symposium on Adaptive Processes*, pp. 179–184, 1974.

[11] M. Lewis, K. Sycara, and P. Walker, *The Role of Trust in Human-Robot Interaction*, pp. 135–159. Springer Int'l Publishing, 2018.

[12] P. Goillau, C. Kelly, M. Boardman, and E. Jeannot, "Guidelines for trust in future ATM systems: Measures," *EUROCONTROL, the European Organisation for the Safety of Air Navigation*, 2003.

[13] M. Madsen and S. Gregor, "Measuring human-computer trust," in *11th Australasian Conf. Information Syst.*, vol. 53, pp. 6–8, Citeseer, 2000.

[14] J.-Y. Jian, A. Bisantz, and C. Drury, "Foundations for an empirically determined scale of trust in automated systems," *Int'l J. Cognitive Ergonomics*, vol. 4, no. 1, pp. 53–71, 2000.

[15] A. P. Vinod, T. H. Summers, and M. M. K. Oishi, "User-interface design for MIMO LTI human-automation systems through sensor placement," in *2016 American Control Conference (ACC)*, pp. 5276–5283, 2016.